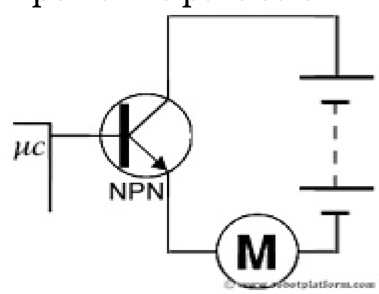
Dual H-bridge Motor Driver - L293D IC

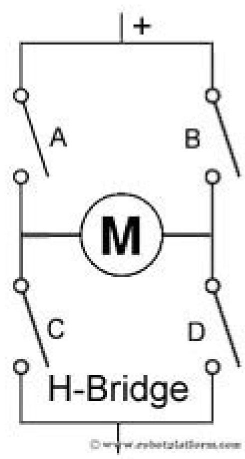
Motor Driver and H-bridge basics

Generally, even the simplest robot requires a motor to rotate a wheel or performs particular action. Since motors require more current then the microcontroller pin can typically generate, you need some type of a switch (Transistors, MOSFET, Relay etc.,) which can accept a small current, amplify it and generate a larger current, which further drives a motor. This entire process is done by what is known as a **motor driver**.



Motor driver is basically a current amplifier which takes a low-current signal from the microcontroller and gives out a proportionally higher current signal which can control and drive a motor. In most cases, a transistor can act as a switch and perform this task which drives the motor in a single direction.

Turning a motor ON and OFF requires only one switch to control a single motor in a single direction. What if you want your motor to reverse its direction? The simple answer is to reverse its polarity. This can be achieved by using four switches that are arranged in an intelligent manner such that the circuit not only drives the motor, but also controls its direction. Out of many, one of the most common and clever design is a H-bridge circuit where transistors are arranged in a shape that resembles the English alphabet "H".

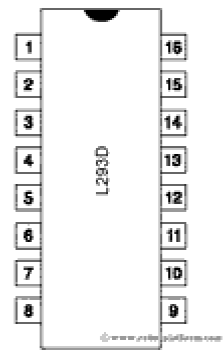


As you can see in the image, the circuit has four switches A, B, C and D. Turning these switches ON and OFF can drive a motor in different ways.

1. Turning on Switches **A** and **D** makes the motor rotate clockwise
2. Turning on Switches **B** and **C** makes the motor rotate anti-clockwise
3. Turning on Switches **A** and **B** will stop the motor (Brakes)
4. Turning off all the switches gives the motor a free wheel drive
5. Lastly turning on **A** & **C** at the same time or **B** & **D** at the same time shorts your entire circuit. So, do not attempt this.

H-bridges can be built from scratch using relays, mosfets, field effect transistors (FET), bi-polar junction transistors (BJT), etc. But if your current requirement is not too high and all you need is a single package which does the job of driving a small DC motor in two directions, then all you need is a L293D IC. This single inexpensive package can interface not one, but two DC motors. L293, L293B and few other versions also does the same job, but pick the L293D version as this one has an inbuilt flyback diode which protects the driving transistors from voltage spikes that occur when the motor coil is turned off.

Introduction to L293D IC



L293D IC generally comes as a standard 16-pin DIP (dual-in line package). This motor driver IC can simultaneously control two small motors in either direction; forward and reverse with just 4 microcontroller pins (if you do not use enable pins). Some of the features (and drawbacks) of this IC are:

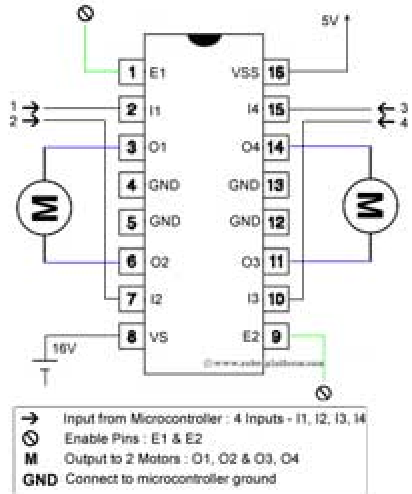
1. Output current capability is limited to 600mA per channel with peak output current limited to 1.2A (non-repetitive). This means you cannot drive bigger motors with this IC. However, most small motors used in hobby robotics should work. If you are unsure whether the IC can handle a particular motor, connect the IC to its circuit and run the motor with your finger on the IC. If it gets really hot, then beware... Also note the words "non-repetitive"; if the current output repeatedly reaches 1.2A, it might destroy the drive transistors.
2. Supply voltage can be as large as 36 Volts. This means you do not have to worry much about voltage regulation.
3. L293D has an enable facility which helps you enable the IC output pins. If an enable pin is set to logic high, then state of the inputs match the state of the outputs. If you pull this low, then the outputs will be turned off regardless of the input states
4. The datasheet also mentions an "over temperature protection" built into the IC. This means an internal sensor senses its internal temperature and stops driving the motors if the temperature crosses a set point
5. Another major feature of **L293D** is its internal clamp diodes. This flyback diode helps protect the driver IC from voltage spikes that occur when the motor coil is turned on and off (mostly when turned off)
6. The logical low in the IC is set to 1.5V. This means the pin is set high only if the voltage across the pin crosses 1.5V which makes it suitable for use in high frequency applications like switching applications (upto 5KHz)
7. Lastly, this integrated circuit not only drives DC motors, but can also be used to drive relay solenoids, stepper motors etc.

L293D Connections

The circuit shown to the right is the most basic implementation of L293D IC. There are 16 pins sticking out of this IC and we have to understand the functionality of each pin before implementing this in a circuit

1. Pin1 and Pin9 are "Enable" pins. They should be connected to +5V for the drivers to function. If they pulled low (GND), then the outputs will be turned off regardless of the input states, stopping the motors. If you have two spare pins in your microcontroller, connect these pins to the microcontroller, or just connect them to regulated positive 5 Volts.

2. Pin4, Pin5, Pin12 and Pin13 are ground pins which should ideally be connected to microcontroller's ground.
3. Pin2, Pin7, Pin10 and Pin15 are logic input pins. These are control pins which should be connected to microcontroller pins. Pin2 and Pin7 control the first motor (left); Pin10 and Pin15 control the second motor(right).
4. Pin3, Pin6, Pin11, and Pin14 are output pins. Tie Pin3 and Pin6 to the first motor, Pin11 and Pin14 to second motor
5. Pin16 powers the IC and it should be connected to **regulated** +5Volts
6. Pin8 powers the two motors and should be connected to positive lead of a secondary battery. As per the datasheet, supply voltage can be as high as 36 Volts.



Truth table

I have shown you where to connect the motors, battery and the [microcontroller](). But how do we control the direction of these motors? Let us take an example:

Suppose you need to control the left motor which is connected to Pin3 (O1) and Pin6 (O2). As mentioned above, we require three pins to control this motor - Pin1 (E1), Pin2 (I1) and Pin7 (I2). Here is the truth table representing the functionality of this motor driver.

Pin 1	Pin 2	Pin 7	Function
High	High	Low	Turn Anti-clockwise (Reverse)
High	Low	High	Turn clockwise (Forward)
High	High	High	Stop
High	Low	Low	Stop
Low	X	X	Stop

High ~+5V, Low ~0V, X=Either high or low (don't care)

In the above truth table you can observe that if Pin1 (E1) is low then the motor stops, irrespective of the states on Pin2 and Pin7. Hence it is essential to hold E1 high for the driver to function, or simply connect enable pins to positive 5 volts.

With Pin1 high, if Pin2 is set high and Pin7 is pulled low, then current flows from Pin2 to Pin7 driving the motor in anti-clockwise direction. If the states of Pin2 and Pin7 are flipped, then current flows from Pin7 to Pin2 driving the motor in clockwise direction.

The above concept holds true for other side of the IC too. Connect your motor to Pin11 and Pin14; Pin10 and Pin15 are input pins, and Pin9 (E2) enables the driver.

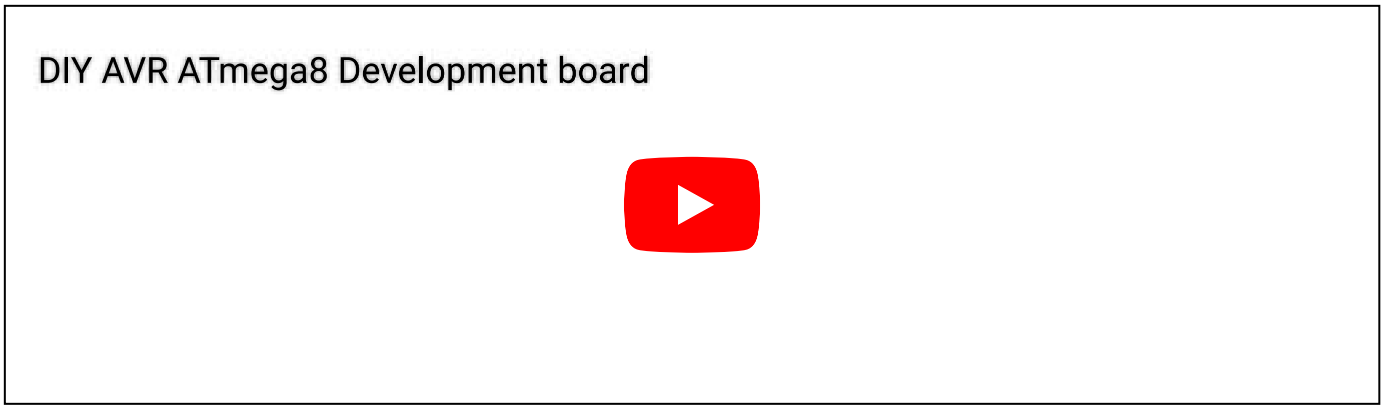
I guess we have already had too much of theory. In the next section, we will start building the board.

Tutorial index:

«Previous - § - Next»

Do you have anything to say?
Visit the Forum to discuss, learn and share anything related to robotics and electronics !!

Featured Videos



Recent Articles

Atmega8 Development Board

A great step-by-step tutorial on building your own Atmel AVR based Atmega8 development board. The board is ideal for beginners with detailed explanation and pictures More...

L293D Motor Driver

For robots to do work, you need to know how to control a motor. L293D is a cleverly packed IC which can control two DC motors in both directions: forwards and reverse. Here is a detailed